

# Aya の 2025 年 UEC 杯アピール文章

まだ実装中で実現できるかは不明ですが

着手に対して大規模言語モデル(LLM, Large Language Model)を使ってコメントを付ける予定です。

SGF に着手のコメント(英語)が付けれた 10000 棋譜(\*1)を使って

1GPU で学習できるサイズの既存の LLM(\*2)に追加学習(LoRA)することで実現させます。

(\*1) SGF の英語のコメント付き棋譜

[https://gtl.xmp.net/reviews/by\\_index?f=1&l=100](https://gtl.xmp.net/reviews/by_index?f=1&l=100)

(\*2) japanese-gpt-neox-3.6b 日本語に特化した 36 億パラメータの GPT 言語モデル

<https://huggingface.co/rinna/japanese-gpt-neox-3.6b>

探索部は以下で昨年と一緒です。

- ・ LeelaZero の棋譜を使って学習させています。

LeelaZero の棋譜 <https://zero.sjeng.org/>

- ・ Policy と Value のみを学習しています。

- ・ ニューラルネットワーク(NN)の構造は 192x15 block の ResNet です。

入力石の配置、ダメ数(1-4 以上)、コウ、直前の位置、シチョウで取られる場所、逃げられない石、です。

- ・ 深層学習以前に利用していた Aya の **playout** も利用しています。探索は MCTS です。

- ・ NN の Value と **playout** の結果を混ぜたものを MCTS の末端で評価値として返します。

**playout** の精度は低く、序盤では 1% の割合でしか使っていません。

終盤(300 手超)でも 20% です。

- ・ 日本ルールは **playout** で対応しています。

中国ルールで最後まで打って、**playout** の途中で 2 回パスしたら補正する手法です。詳細は下を。

日本ルールへの対応(深層学習登場以前の **playout** で解決する手法)

[http://www.yss-aya.com/bbs\\_log/bbs2015.html#bbs139](http://www.yss-aya.com/bbs_log/bbs2015.html#bbs139)

- ・ **playout** を利用した Aya の構造については下の書籍に書いています。

「コンピュータ囲碁 モンテカルロ法の理論と実践」松原 仁編・美添 一樹・山下 宏著、2012 年

<http://www.yss-aya.com/book2011/>

# Aya's 2025 UEC Cup appeal

While it's still in the planning stages and it's unclear if this will work,

We plan to use a large-scale language model (LLM) to annotate moves.

We will achieve this by using 10,000 Go games (\*1) with move comments (in English) in SGF.

and additional training (LoRA) on an existing LLM (\*2) that is small enough to train on a single GPU.

(\*1) SGF game records with English comments

[https://gtl.xmp.net/reviews/by\\_index?f=1&l=100](https://gtl.xmp.net/reviews/by_index?f=1&l=100)

(\*2) japanese-gpt-neox-3.6b, A 3.6 billion parameter GPT language model specialized for Japanese

<https://huggingface.co/rinna/japanese-gpt-neox-3.6b>

Search and NN is the same as last year, as shown below.

1. I use LeelaZero game records(SGF) to train Neural Network(NN).

LeelaZero games. <https://zero.sjeng.org/>

2. Policy and Value are learned.

3. NN structure is 192x15 block ResNet.

Inputs are Stones, Liberties(1,2,3,4>=),Ko, Previous move position, Ladder.

4. Aya uses playout which work since 2008 to 2017, before Deep Learning . Search is MCTS.

5. Return mix score using NN value and playout result at terminal node of MCTS.

Playout is not trustable, so mixing rate is only 1% in opening game, and 20% in endgame.

6. Only playout handle Japanese rule.

After playout plays at last in Chinese rule, score is adjusted if there are two consecutive passes. Details are

Japanese rule in MCTS (to solve using playout before deep learning). In Japanese

[http://www.yss-aya.com/bbs\\_log/bbs2015.html#bbs139](http://www.yss-aya.com/bbs_log/bbs2015.html#bbs139)

7. I wrote Aya's playout in this book.

"Computer Go:Theory and Practice of Monte Carlo Method". Hitoshi Matsubara, Kazuki Yoshizoe, Hiroshi Yamashita, (2012)

<http://www.yss-aya.com/book2011/>